Homomorphic encryption functions and cryptographic protocols (Algebra, Languages and Computation)

AUTHOR(S):

Yamamura, Akihiro; Jajcayova, Tatiana; Kurokawa, Takashi

数理解析研究所講究録 1437 巻 2005 年 97-106

Homomorphic encryption functions and cryptographic protocols

Akihiro Yamamura* Tatiana Jajcayova† Takashi Kurokawa*

1 Introduction

We propose efficient oblivious transfers and private information retrieval schemes using a homomorphic encryption. Our private information retrieval enables the user to retrieve block data from the database consisting of several binary strings without iteration. This paper is an extended abstract and the detailed version will be published elsewhere.

A mapping between algebraic systems is called a *homomorphism* if it preserves the algebraic structures. In cryptography, trapdoor one-way homomorphism between cyclic groups have been proposed and applied to numerous secure protocols, for example, secret ballot elections schemes. Such encryptions include ElGamal, Goldwasser-Micali, Benaloh, Okamoto-Uchiyama, Paillier cryptosystems and so on.

Oblivious Transfer (OT) refers to several types of two party protocols, where one party, the sender, transmits part of its input to another party, the chooser, in a way that protects both parties: the sender is assured that the chooser does not get more information than it is entitled, and chooser is assured that the sender does not learn which part of the inputs it received. The notion of *1-out-of-2 oblivious transfer* (OT_1^2 for short) was introduced in [4], as generalization of Rabin's concept of OT [10].

Private Information Retrieval (PIR) *schemes* [2] allow a user to access a database consisting of N data $m_1, m_2, \ldots, m_N$ (usually data are just a bit) and read any elements without a database manager learning which element was accessed. PIR schemes do not protect the owner of the database, because they do not prevent the user from learning more than a single element. Currently, the question of protecting the database was addressed as well. A

*National Institute of Information and Communications Technology
†Comenius University, Slovakia

PIR scheme where a user does not learn more than a single data is called a *Symmetric* PIR (SPIR) [5].

An oblivious transfer can be implemented under different assumptions. In this paper, we construct an efficient OT_1^N based on the p-subgroup assumption and a PIR in which database consists of block data based on the subgroup membership problem. These are constructed under Okamoto-Uchiyama encryption [9]. Unifying these constructions, a new efficient SPIR based on the p-subgroup assumption. Okamoto-Uchiyama encryption is a public key cryptosystem whose security us based on the so-called *p-subgroup assumption.*

2 Okamoto and Uchiyama's encryption scheme

2.1 Preliminaries

Okamoto and Uchiyama [9] introduces a new trapdoor one-way function based on the hardness of factorization of composites of the form p^2q. Their scheme has many interesting properties.

Suppose that p is a prime number of size k. Let Γ be the p-Sylow subgroup of $(\mathbb{Z}/(p^2))^*$, that is, Γ is the maximal subgroup whose order is a power of p. The group $(\mathbb{Z}/(p^2))^*$ has order $\phi(p^2) = p(p-1)$. Thus $(\mathbb{Z}/(p^2))^*$ is an internal direct product of Γ and the subgroup of order $p-1$. It is easy to see that the subgroup of order $p-1$ is isomorphic to $(\mathbb{Z}/(p))^*$ and so it is cyclic. On the other hand, Γ has order p and so it is cyclic. It follows that $(\mathbb{Z}/(p))^*$ is cyclic because p and $p-1$ are coprime.

We next show that if $x \equiv 1(\mathrm{mod}p)$, then we have $x^p \equiv 1(\mathrm{mod}p^2)$. Suppose $x \equiv 1(\mathrm{mod}p)$. Then $x = cp + 1$ for some c in $\mathbb{Z}$. We have $x^p = (cp+1)^p = \sum_{i=0}^{p} \binom{p}{i} (cp)^{p-i}$. Hence, $x^p = dp^2 + 1$ for some d in $\mathbb{Z}$. It follows that $x^p \equiv 1(\mathrm{mod}p^2)$. It follows that $\Gamma = \{x \in (\mathbb{Z}/(n))^* | x \equiv 1(\mathrm{mod}p)\}$.

A homomorphism L of Γ into the additive group $(\mathbb{Z}/(p), +)$ is defined by $L(x) = \frac{x-1}{p}$ for every x in Γ. Then L is an isomorphism of Γ onto $\mathbb{Z}/(p)$, that is, we have $L(ab) \equiv L(a) + L(b)(\mathrm{mod}p)$ for a, b in Γ (see [9]). In particular, we have $L(y) = mL(x)$ for every x, y in Γ with $y = x^m$ $(m \in \mathbb{Z})$. Hence, $m = \frac{L(y)}{L(x)}$ unless $L(x) = 0$. Note that $L(x) = 0$ if and only if x is the identity element of Γ.

Suppose that $n = p^2q$, where p and q are primes of same k. Recall that Γ is the p-Sylow subgroup of $(\mathbb{Z}/(p^2))^*$. In fact we can consider that Γ is the p-Sylow subgroup of $(\mathbb{Z}/(n))^*$. The group $(\mathbb{Z}/(p^2))^*$ is abelian

and so it is a direct product of its Sylow subgroups. Note that Sylow subgroups are uniquely determined in an abelian group. By the chinese remainder theorem, we have $(\mathbb{Z}/(n))^* = (\mathbb{Z}/(p^2))^* \times (\mathbb{Z}/(q))^*$. Let π_1 be the projection into the first component, that is, π_1 is the mapping of $(\mathbb{Z}/(n))^*$ into $(\mathbb{Z}/(p^2))^*$ defined by $\pi_1(x(\mathrm{mod}n)) = x(\mathrm{mod}p^2)$. We define $\psi_{p-1} : (\mathbb{Z}/(n))^* \to (\mathbb{Z}/(n))^*$ by $\psi_{p-1}(x(\mathrm{mod}n)) = x^{p-1}(\mathrm{mod}n)$. We define a subgroup U to be the complement of Γ in $(\mathbb{Z}/(p^2))^*$, that is, U is a subgroup of $(\mathbb{Z}/(p^2))^*$ such that $(\mathbb{Z}/(p^2))^* \cong \Gamma \times U$. The order of U is $p-1$. Then we have $(\mathbb{Z}/(n))^* = \Gamma \times U \times (\mathbb{Z}/(q))^*$.

2.2 Okamoto-Uchiyama encryption

Let p, q be primes such that $|p| = |q| = k$. Set $n = p^2q$. We should note that $|n| = 3k$. Take an element g randomly and uniformly from $(\mathbb{Z}/(n))^*$ so that the order of $g_p = \pi_1(\psi_{p-1})(g)(= g^{p-1} \bmod p^2)$ is p. Let $h = g^n(\mathrm{mod}n)$.

Public key The public key is (n, k, g, h).

Secret key The secret key is (p, q).

Encryption Suppose m is a plaintext with $0 < m < 2^{k-1}$. Hence, the length of plaintexts is bounded by k. Choose r randomly and uniformly from $\mathbb{Z}/(n)$. Then the plaintext is encrypted by $C = g^m h^r(\mathrm{mod}n)$

Decryption Bob computes $m = \frac{L(C^{p-1} \bmod p^2)}{L(g^{p-1} \bmod p^2)} (\mathrm{mod}\, p)$. Note $\pi_1(\psi_{p-1}(C))$ belongs to Γ.

It is shown in [9] that the hardness of inverting the encryption function is equivalent to the hardness of factoring the composites of the type p^2q and the encryption scheme is semantic secure if and only if the *p-subgroup problem* is intractable.

2.3 p-subgroup assumption

Let $\mathcal{G}$ be an instance generator for the Okamoto-Uchiyama encryption scheme; $\mathcal{G}$ is a probabilistic algorithm that outputs (n, g, C) for the input 1^k and then (n, g, k) is a public key and C is a ciphertext of the message 0 or 1 (say b), taht is, $C = g^m h^r$. The *p-subgroup problem* is intractable if for any (uniform/non-uniform) probabilistic algorithm $\mathcal{A}$, any constant c, we have

$$\mathrm{Prob}(\mathcal{A}(1^k, n, g, m_0, m_1, C) = b < \frac{1}{2} + \frac{1}{k^c}$$

for sufficiently large k. The *p-subgroup assumption* is to assume that the p-subgroup problem is intractable.

3 Subgroup membership problem

Let G be a group, and let H be its subgroup. The subgroup membership problem is to decide whether or not a given element $g \in G$ belongs to H (see [12, 13]). We suppose that every element in G has a binary representation of size k, where k is the security parameter.

The predicate for the membership of a subgroup is denoted by Mem, that is, $\mathrm{Mem}(G, H, x) = 1$ if $x \in H$ and $\mathrm{Mem}(G, H, x) = 0$ if $x \in S = G \setminus H$. The *subgroup membership problem* is to compute Mem in polynomial time in k when we inputs 1^k to an instance generator $\mathcal{IG}$ and obtain a pair of groups (G, H) and an element g in G, which is uniformly and randomly chosen from H or G according to the coin toss $b \stackrel{R}{\leftarrow} \{0, 1\}$. If there does not exist a probabilistic polynomial time algorithm that computes Mem with a probability substantially larger than $\frac{1}{2}$, then we say that the membership problem is *intractable.*

Examples Some cryptographic assumptions are characterized as a subgroup membership problem. Let G be the subgroup of $(\mathbb{Z}/(N))^*$ consisting of the elements whose Jacobi symbol is 1. Then QR is the membership problem of $H = \{x \in G \mid x = y^2 \bmod N \text{ for } y \in (\mathbb{Z}/(N))^*\}$ in the group G.

Suppose G is a cyclic group generated by x. The DDH is the problem that given $[x, x^a, x^b, x^c]$ decides whether $x^{ab} = x^c$ or not. Then DDH is the membership problem of the subgroup $< (x, x^a) >$ in the group $C \times C = \{(x^e, x^f) \mid 0 \leq e, f < |x|\}$. For other examples, see [8, 12, 13].

Theorem 3.1 *(1) The p-subgroup assumption is stronger than the subgroup membership assumption of $U \times (\mathbb{Z}/(q))^*$ in $(\mathbb{Z}/(n))^*$, that is, the p-subgroup problem is reduced to the subgroup membership problem.*
(2) The subgroup membership assumption for $U \times (\mathbb{Z}/(q))^*$ in $(\mathbb{Z}/(n))^*$ is stronger than assuming intractability of factorization of composites of the form p^2q, that is, the subgroup membership problem can be reduced to the factorization of p^2q.

4 Oblivious transfer

The sender $\mathcal{S}$ has the secret data $m_1, m_2, \ldots, m_N$. Set $X = (m_1, m_2, \ldots, m_N)$. The receiver $\mathcal{R}$ wishes to obtain one of the data (say m_α) and send a query to $\mathcal{S}$ so that $\mathcal{S}$ cannot obtain any information on α while $\mathcal{R}$ gets only m_α. Thus $\mathcal{R}$ does not obtain any information on the other data. This requirement makes the difference from private information.

4.1 Computationally secure OT_1^N

A general one round oblivious transfer protocol runs as follows:

Step 1 $\mathcal{R}$ generates the system parameters. $\mathcal{R}$ computes a query $\mathrm{Query}(\alpha)$ using his random tape (coin toss), which $\mathcal{R}$ keeps secret. Then $\mathcal{R}$ sends $\mathrm{Query}(\alpha)$ to $\mathcal{S}$.

Step 2 $\mathcal{S}$ receives $\mathrm{Query}(\alpha)$. He performs a polynomial-time computation for the input X, $\mathrm{Query}(\alpha)$ and his random tape. The computation yields the answer $\mathrm{Answer}(\mathrm{Query}(\alpha))$. $\mathcal{S}$ sends $\mathrm{Answer}(\mathrm{Query}(\alpha))$ back to $\mathcal{R}$.

Step 3 $\mathcal{R}$ receives $\mathrm{Answer}(\mathrm{Query}(\alpha))$. He performs a polynomial-time computation using $\mathrm{Answer}(\mathrm{Query}(\alpha))$ and his private information. The computation yields the αth data m_α of X.

Correctness
If both party play honestly (no cheating), $\mathcal{R}$ obtains m_α for any sequence X of data and any query $\mathrm{Query}(\alpha)$.

Privacy for $\mathcal{R}$
$\mathcal{S}$ cannot distinguish a query for the αth and the βth data for all α and β. In our scheme, the privacy for $\mathcal{R}$ is computational; $\mathcal{S}$ cannot distinguish two queries from $\mathcal{R}$ by a polynomial time (probabilistic) computation with non-negligible probability. Formally, for all constants c, for all database of length n, for any two $1 \leq \alpha, \beta \leq N$, and any (uniform/non-uniform) probabilistic algorithm $\mathcal{A}$, there exists an integer K such that for all $k > K$ we have

$$|\mathbf{Prob}(\mathcal{A}(\mathrm{Query}(\alpha)) = 1) - \mathbf{Prob}(\mathcal{A}(\mathrm{Query}(\beta)) = 1)| < \sigma \ ,$$

where k is the security parameter of the protocol and $\sigma = \frac{1}{(\mathrm{Max}(k,n))^c}$.

Privacy for $\mathcal{S}$
$\mathcal{R}$ cannot obtain any information on the other data. In our scheme, the privacy for $\mathcal{S}$ is unconditional, that is, $\mathcal{R}$ cannot obtain any partial information of m_β for all $\beta \neq \alpha$ even with unlimited computing power.

We do not take into consideration active attacks of $\mathcal{R}$ or $\mathcal{S}$ in this paper. So we suppose that both party $\mathcal{R}$ and $\mathcal{S}$ are honest and follow the protocol.

Computation
Computations of both $\mathcal{R}$ and $\mathcal{S}$ are bounded above by a polynomial in the size N of the database and the security parameter k.

4.2 Proposed scheme

We construct a 1-out-of-N one round oblivious transfer OT_1^N using Okamoto-Uchiyama encryption scheme. Suppose that the sender $\mathcal{S}$ has N data m_1, m_2, ..., m_N and the receiver $\mathcal{R}$ wishes to know the αth data m_α. The security parameter k is taken large enough; we take k so that $0 \leq m_i < 2^{k-1}$ for every i.

Step 1 $\mathcal{R}$ generates primes p and q with $|p| = |q| = k$. We may assume $N \leq 2^{k-1}$, otherwise any computation related transaction with $\mathcal{R}$ takes more than polynomial time. Set $n = p^2q$. We use the same notation as Section 2. $\mathcal{R}$ takes an elements g_1 randomly and uniformly from $(\mathbb{Z}/(n))^*$ so that $|g_1^{p-1} (\mathrm{mod} p^2)| = p$. Let g_2 to be g_1^d, where d is randomly and uniformly chosen from $\{1, 2, \ldots, p-1\}$. $\mathcal{R}$ also chooses u randomly and uniformly. Set $f = g_2^{n(u+1)-\alpha}$. The query for the αth data (Query(α) for short) is defined to be (n, k, g_1, g_2, f). $\mathcal{R}$ sends Query(α) to $\mathcal{S}$.

Step 2 $\mathcal{S}$ chooses randomly and uniformly r_i and s_i from $\mathbb{Z}/(n)$ for every $i = 1, 2, \ldots, N$. Then $\mathcal{S}$ computes $c_i = g_1^{m_i + ns_i}(fg_2^i)^{r_i}$ for every $i = 1, 2, \ldots, N$. The answer (denoted by Answer(Query(α))) consists of $(c_1, c_2, \ldots, c_N)$. $\mathcal{S}$ sends Answer(Query(α)) to $\mathcal{R}$.

Step 3 $\mathcal{R}$ computes

$$\frac{L(\pi_1(\psi_{p-1}(c_\alpha)))}{L(\pi_1(\psi_{p-1}(g_1)))} \bmod p = \frac{L(c_\alpha^{p-1})}{L(g_1^{p-1})} (\mathrm{mod} p)$$

and obtain m_α.

Correctness We have $c_\alpha = g_1^{m_\alpha + ns_\alpha}(fg_2^\alpha)^{r_\alpha} = g_1^{m_\alpha + ns_\alpha}(g_2^{n-\alpha} g_2^{nu} g_2^\alpha)^{r_\alpha} = g_1^{m_\alpha + ns_\alpha}(g_2^{n(u+1)})^{r_\alpha} = g_1^{m_\alpha + ns_\alpha} g_2^{nr_\alpha(u+1)}$. Since the order of $(\mathbb{Z}/(p^2))^*$ is $p(p-1)$, $g_1^{n(p-1)}$ $(= (g_1^{pq})^{p(p-1)})$ is the identity element in $(\mathbb{Z}/(p^2))^*$. Hence, we have $(g_2^{nr_\alpha(u+1)})^{p-1} \equiv 1 (\mathrm{mod} p^2)$ and $(g_1^{ns_\alpha})^{p-1} \equiv 1 (\mathrm{mod} p^2)$. Then it follows that $\frac{L(c_\alpha^{p-1})}{L(g_1^{p-1})} = \frac{L((g_1^{p-1})^{m_\alpha})}{L(g_1^{p-1})} = m_\alpha$. Hence, $\mathcal{R}$ can correctly obtain m_α.

Privacy We shall show that the OT_1^N has computational security for $\mathcal{R}$' privacy and unconditional security for $\mathcal{S}$'s privacy against passive attacks. For active attacks, it is necessary to integrate zero-knowledge proof or so.

Theorem 4.1 *The privacy for $\mathcal{R}$'s choice α is equivalent to the p-subgroup assumption.*

Theorem 4.2 *The privacy for $\mathcal{S}$ is unconditionally secure against passive attacks, that is, an honest $\mathcal{R}$ cannot get any information on m_i for any i $(i \neq \alpha)$.*

5 Private information retrieval

5.1 Computationally secure PIR with a single database server

We review the general scheme of a computational one-server private information retrieval (PIR for short) scheme. A computational PIR scheme [3] is a protocol for two players, a user $\mathcal{U}$ and a database manager $\mathcal{DB}$. Both are able to perform only probabilistic polynomial time computation. $\mathcal{DB}$ maintains a database, which is a sequence $X = m_1, m_2, \ldots, m_N$ of binary string of the same size. We note that each data has usually length 1 for a PIR scheme, that is, $m_1, m_2, \ldots, m_N \in \{0, 1\}$ and that if $\mathcal{U}$ wants to obtain more than one bit, he needs to iterate the PIR protocol. The goal of the protocol is to allow $\mathcal{U}$ to obtain the αth bit string m_α without leaking any information on m_α to $\mathcal{DB}$. We here construct a PIR scheme that allows $\mathcal{U}$ to get a bit string of fixed size by only one procedure. The protocol runs as follows:

Step 1 The user $\mathcal{U}$ generates the system parameters and computes a query Query(α) using his random tape (coin toss), which $\mathcal{U}$ keeps secret. Then he sends Query(α) to $\mathcal{DB}$.

Step 2 $\mathcal{DB}$ receives Query(α). He performs a polynomial-time computation for the input X, Query(α) and his random tape. The computation yields the answer Answer(Query(α)). He sends Answer(Query(α)) back to $\mathcal{U}$.

Step 3 $\mathcal{U}$ receives Answer(Query(α)). He performs a polynomial-time computation using the answer Answer(Query(α)) and his private information (his random tape). The computation yields the αth bit m_α of the database.

Correctness
For any database sequence X and for any query Query(α) for αth information of X, $\mathcal{U}$ obtains m_α at the end.

Privacy for $\mathcal{U}$
$\mathcal{DB}$ cannot distinguish a query for the αth and the βth data for all α and β by a polynomial-time (probabilistic) computation with non-negligible probability. Formally, for all constants c, for all database sequences X of length

n, for any two $1 \leq \alpha, \beta \leq n$, and any (uniform/non-uniform) probabilistic algorithm $\mathcal{A}$ there exists an integer K such that for all $k > K$ we have

$$|\mathbf{Prob}(\mathcal{A}(\mathrm{Query}(\alpha)) = 1) - \mathbf{Prob}(\mathcal{A}(\mathrm{Query}(\beta)) = 1)| < \sigma \ ,$$

where k is the security parameter of the protocol and $\sigma = \frac{1}{(\mathrm{Max}(k,n))^c}$.

Computation
Computations of both $\mathcal{DB}$ and $\mathcal{U}$ are bounded above by a polynomial in the size N of the database and the security parameter k.

5.2 Proposed scheme

Step 1 Suppose that $\mathcal{U}$ wants to obtain the αth information m_α. $\mathcal{U}$ generates the query consisting of $g_1, g_2, g_3, \ldots, g_N$, where g_i are elements of $(\mathbb{Z}/(n))^*$ such that $g_\alpha \notin U \times (\mathbb{Z}/(q))^*$ and $g_i \in U \times (\mathbb{Z}/(q))^*$ unless $i = \alpha$ for every $i = 1, \ldots, N$.

Step 2 The answer Answer(Query(α)) is defined to be $h_0^r g_1^{m_1} g_2^{m_2} g_3^{m_3} \cdots g_N^{m_N}$, where $h_0 = g_j^n (\mathrm{mod} n)$ with $\mathcal{DB}$'s choice of j and r is randomly and uniformly chosen from $\mathbb{Z}/(n)$. $\mathcal{DB}$ sends Answer(Query(α)) to the user.

Step 3 $\mathcal{U}$ retrieves m_α by computing $\frac{L((\mathrm{Answer}(\mathrm{Query}(\alpha)))^{p-1} \bmod p^2)}{L(g_\alpha^{p-1} \bmod p^2)} \bmod p$.

Correctness We note that $(g_j^{m_i})^{p-1} \equiv 1 (\mathrm{mod} p^2)$ unless $i = \alpha$ and $h_0^r \equiv 1 (\mathrm{mod} p^2)$. Thus we have

$$\frac{L(\mathrm{Answer}(\mathrm{Query}(\alpha)))}{L(g_\alpha^{p-1})} = \frac{L(h_0^r g_1^{m_1} g_2^{m_2} g_3^{m_3} \cdots g_N^{m_N})}{L(g_\alpha^{p-1})} = \frac{L((g_\alpha^{m_\alpha})^{p-1})}{L(g_\alpha^{p-1})} = m_\alpha.$$

Privacy The privacy is guaranteed if the subgroup membership problem of $U \times (\mathbb{Z}/(q))^*$ in $(\mathbb{Z}/(n))^*$ is intractable. The proof is almost same as the security proof of the PIR scheme based on the subgroup membership problem in [12, 13] and so we omit the proof.

6 Discussion

Such encryptions include ElGamal, Goldwasser-Micali, Benaloh, Okamoto-Uchiyama and Paillier cryptosystems share many similarities, no satisfactorily uniform mechanism of homomorphic encryptions has been explained so far. It is quite interesting to study homomorphic encryptions in terms of

group theory. Our future work is to study mechanisms, constructions and semantic security of homomorphic encryptions and to construct homomorphic encryptions on non-cyclic abelian groups.

References

[1] C. Cachin, S. Micali, M. Stadler, Computationally Private Information Retrieval with Polylogarithmic Communication, Eurocrypt'99 LNCS, Vol. 1592. Springer-Verlag, (1999) 402–414.

[2] B. Chor, O. Goldreich, E. Kushilevitz and M. Sudan, Private Information Retrieval, FOCS'95 (1995) 41–50.

[3] B. Chor, N. Gilboa, Computationally Private Information Retrieval STOC'97 (1997) 304–313.

[4] S. Even, O. Goldreich and A. Lempel, A randomized protocol for signing contracts, Communications of the ACM 28 (1985) 637–647.

[5] Y. Gertner, Y. Ishai, E. Kushilevitz, T. Malkin, Protecting data privacy in private data information retrieval schemes, STOC'98 (1998) 151–160.

[6] E. Kushilevitz and R. Ostrovsky, One-Way Trapdoor Permutations Are Sufficient for Non-trivial Single-Server Private Information Retrieval, Eurocrypt 2000 LNCS, Vol. 1807. Springer-Verlag, (2000) 104–121.

[7] M. Naor and B. Pinkas, Oblivious transfer and polynomial evaluation, STOC'99 (1999) 245–254.

[8] J. Nieto, C. Boyd and E. Dawson, A public key cryptosystem based on the subgroup membership problem, ICICS 2001 (2001) 352-363.

[9] T. Okamoto, S. Uchiyama, A New Public-key Cryptosystem as Secure as Factoring, Eurocrypt'98 LNCS, Vol. 1403. Springer-Verlag, (1998) 308–318.

[10] M. Rabin, How to exchange secerts by oblivious transfer, Technical Report TR-81, Harvard University (1981)

[11] W. Tzeng, Efficient 1-Out-n Oblivious Transfer Schemes, PKC 2002, LNCS, Vol. 2274. Springer-Verlag, (2002) 159–171.

[12] A. Yamamura, T. Saito, Private Information Retrieval Based on teh Subgroup Membership Problem, ACISP 2001 LNCS, Vol. 2119. Springer-Verlag, (2001) 206–220.

[13] A. Yamamura and T. Saito, Subgroup membership problems and applications to information security, Scientiae Mathematicae Japonicae, (1) **57** (2003) 25–41.

[14] A.Yamamura, T,Jajcayova and T.Kurokawa, Oblivious transfer and private information retrieval based on the p-subgroup assumption, SOFSEM 2005, Communications, Slovak Society for Computer Science, (2005) 101-110.